processes or both, and alerting a human user or an intelligent software that are authorized to decide upon a further action.

[0094] Upon receiving positive symptoms from probe **26**, in block **296** the decision unit, by means of the sensor main unit **25** and communication unit **24** initiates an alert and optionally also notifies fellow agents **28**.

[0095] Whenever the received symptoms are fuzzy, or non-decisive, then in step **292** the sampled symptoms are received from probe **26** and are evaluated and their weight is considered in step **294**, by comparing them to a pre established threshold level. Typically, the weights are initially set to equal values (1.0's); when the user confirms an alert, the weights of exitatory inputs are incremented while the weights of inhibitory inputs are decremented, and vice versa. This simple learning mechanism, however, is implemented by a separate module. Also, the user may explicitely set the weights to some reasonable values. If a comparison shows that the threshold level is not met, then in step **295** the sensor main unit **25** orders probe **26** to release the frozen process and resume the monitoring, and no alert is initiated. If, however, the threshold level is met, the procedure continues to block **296**, in which the sensor main unit **25** initiates an alert, orders probe **26** to manipulate the suspect signal, and optionally transmit an alert to fellow agents **28**.

[0096] When applicable, unit **25** may further maintain a 'sand box', or a 'redo buffer' (which are respectively commercial names for mechanisms that put the suspect into a secured environment, or record a suspected sequence of actions so that they may be inverse) or deceive the offender to think that it is still performing unnoticed.

[0097] In a preferred embodiment of the invention the sensor of the invention is a learning unit, that accumulates information from several sources. More particularly, its knowledge base **150** is dynamically updated as a result of the tests that are made, or from information obtained from external sources, or from the user himself.

[0098] The present invention deals with several typical OS offenders, as follows:

[0099] passive offender: eavesdrops on signals passed between the OS and another process. It plants its SCR in the extensible chain of handlers of a stack, and then simply waits for the OS to route signals to that SCR. When the SCR receives such a signal from the OS, it can push it into a shared memory area that is available for the originating process of the offender, that process would typically cache the information and send it out later through some output device or communication port.

[0100] direct approach passive offender: operates after the SCR is implanted. It directly sends out eavesdropped information, or manipulates signals before passing them on, all without involving the SCR's originating process. This behavior implies that logged output actions would go under the identity of the offended process. on the other hand, this direct approach may result in both noticeable degradation and anomalies in the performance of the offended process.

[0101] active offender: takes hold of the offended process main logic (or a 'subclass' of it). It plants its SCR in the extensible chain, and then deliberately initiates a signal that triggers the SCR into action, rather than wait for the OS to pass such a signal. When the SCR receives that signal, it would transplant a predefined wrapper on the offended process' relevant procedure, forcing a new behavior. Except for the initiating signal, this type of activity would typically go on without intervention of the initiating offender process, as with the 'direct approach' offender.

[0102] The invention provides several examples of sensors that can be implemented in some known in the art operating systems.

[0103] public sensor mechanism: a public sensor, according to the present invention, is an independent process, preferably a program that runs by itself directly under the OS, and exists for the purpose of handling periodic service requests that the system expects to receive (i.e., daemon as in the Unix OS or a service) to ensure its continuous availability. The main unit (i.e., main unit **25**) of the public sensor mechanism plants its probe **26** (implemented as an SCR) into the stack chain (e.g., Winsock) in much the same way that an offender does. Public sensor mechanism is capable of protecting various stack types with minimal or no preparations on their part, however, each stack requires its specific probe. It also pushes its own identifier onto the probe's shared memory section, for a later use. The 'identifier' in this case is a unique string or number that enables the probe to distinguish the relevant public sensor from other modules. The identification is typically provided by the OS.

[0104] private sensor mechanism: a private sensor is implemented as a part of an input-element (such as an HTML input tag, which is a code that informs the web browser how to display information) that is protected. The enhanced input-element is either available to developers of compiled programs before compilation, or replaces the standard component/library in case of scripting programs and authoring environments (like an HTML input tag). In that case, the load on the system performance is minimized since protection is applied only when actually needed. Sensor **41** as shown in **FIG.** 2B, is an example of a private sensor.

[0105] The following is a description of a first operation mode of a public sensor for first setting with passive offender, according to an embodiment of the invention:

[0106] In the first operation mode, the sensor has to detect and evaluate whether a new code module was inserted into the context of a specific extensible process.

[0107] FIGS. **4**A-**4**E are flow diagrams describing several optional verification tests that are performed by the sensor of the invention in order to detect illegal memory breach by means of shared codes resources.

[0108] In the Embodiment of **FIG. 4**A:

[0109] In some cases, the Operating System enables an on-line identification of the occurrence of adding an SCR to a process during the process operation. The embodiment of **FIG. 4**A, is applicable for the case when the operating system enables obtaining a list of SCRs mapped to given processes. The procedure therefore checks the available list, and if a new, suspected SCRs is found within the list, an alert is issued.